What is claimed is:

1. A method for optimizing clustering in a design structure matrix comprising the steps of:

applying at least one genetic operator to a parent population of design structure matrix clusterings to produce an offspring population of design structure matrix clusterings;

using a scoring metric to score each of said offspring population of design structure matrix clusterings;

terminating the method if a termination condition has been satisfied and defining an optimal design structure matrix clustering; and,

performing a selection operation to generate a new parent population of design structure matrix clusterings if said termination condition has not been satisfied and repeating said steps of applying at least one genetic operator to said new parent population to generate a new offspring population, using a scoring metric to score said new offspring population, and terminating the method if a termination condition has been satisfied, until said termination condition is satisfied.

2. A method as defined by claim 1 wherein the step of performing a selection operation includes replacing at least a portion of said parent population of design structure matrix clusterings with a high scoring portion of said offspring population of design structure matrix clusterings.

3. A method as defined by claim 1 wherein said at least one genetic operator is one or more of a crossover operator and a mutation operator.

4. A method as defined by claim 1 wherein said at least one genetic operator is one or more of a probabilistic operator, a distribution estimation operator, and a stochastic search operator.

5. A method as defined by claim 1 wherein the step of applying at least one genetic operator comprises applying said at least one genetic operator according to a probability.

6. A method as defined by claim 1 wherein the step of determining if a termination condition has been satisfied comprises determining if a particular number of iterations of said steps of applying a genetic operator to said parent population of design structure matrix clusterings and of scoring said offspring population of design structure matrix clusterings have been performed.

7. A method as defined by claim 1 wherein said scoring metric conveys at least two categories of information including cluster complexity and cluster accuracy.

8. A method as defined by claim 1 wherein said scoring metric comprises a minimum description length scoring metric.

9. A method as defined by claim 8 wherein the step of using said minimum description length scoring metric comprises:

using said minimum description length scoring metric to describe each of said plurality of design structure matrix clusterings in said offspring population of design structure matrix clusterings as a sum of a first component that describes the size of a data structure required to represent said design structure matrix clustering and a second component that describes the size of a data structure required to describe mismatched nodes from said design structure matrix clustering.

10. A method as defined by claim 9 and further including the step of weighting said first and second components.

11. A method as defined by claim 8 wherein each of said offspring population of design structure matrix clusterings is represented as:

$$\sum_{i=1}^{n_c} \log n_c + (\log n_c) \times cl_i$$

where n_c is the number of clusters in said each of said offspring population of design structure matrix clusterings, n_n is the number of nodes, and cl_i is the number of nodes in the i^{th} cluster, and log is based on 2.

12. A method as defined by claim 1 wherein at least one of said offspring population of design structure matrix clusterings has at least two clusters that overlap with one another.

13. A method as defined by claim 1 wherein at least one of said offspring population of design structure matrix clusterings has a bus.

14. A method as defined by claim 1 and further including an initial step of developing the design structure matrix.

15. A method as defined by claim 1 and further including an initial step of generating said parent population of design structure matrix clusterings through random clustering.

16. A method as defined by claim 1 and further including the step of developing said parent population of design structure matrix clusterings by representing interactions between the i-th and j-th gene in a population using the relationship:

$$s_{ij}^t = \left| f_{ai=o,aj=1}^t - f_{ai=0,aj=0}^t - f_{ai=1,aj=1}^t + f_{ai=1,aj=0}^t \right|$$

where $f_{ai=x,aj=y}^t$ is the fitness value where the i=th gene is x and the j-th gene is y, and t is the generation;

defining a set $D = \{s_{ij}^t | s_{ij}^t \text{ is } \textit{defined}\}$ where

$$s_{ij} = \frac{1}{|D|} \sum_{t=1, s_{ij}^t \in D}^{T} s_{ij}^t$$

where T is the current generation; and

wherein said parent population of clustered design structure matrix models are binary and have a plurality of cell $d_{i\,j}$ that are assigned binary values according to a threshold θ:

$$d_{i\,j} = \{0 \text{ if } s_{ij} \leq \theta; \text{ and } 1 \text{ if } s_{ij} \geq \theta\}.$$

17. A method as defined by claim 16 and further including the step of assigning said threshold θ according to a two-mean algorithm.

18. A method as defined by claim 1 wherein said termination condition occurs when said offspring population of design structure matrix clusterings converges, and wherein the converged offspring population of design structure matrix clusterings is defined as said optimal design structure matrix clustering.

19. A method as defined by claim 1 wherein the step of defining said optimal design structure matrix clustering includes selecting the highest scoring of said offspring population of design structure matrix clusterings.

20. A method as defined by claim 1 and further including the step of using said optimal design structure matrix clustering to organize a population of variables into modules.

21. A method as defined by claim 20 wherein said at least one genetic operator is at least one first genetic operator, and wherein the method further includes the step of applying at least one second genetic operator to said population of variables on a module-specific basis.

22. A method as defined by claim 21 wherein the step of applying said at least one second genetic operator and the steps of claim 1 are performed iteratively until a second termination condition is satisfied.

23. A method as defined by claim 21 and further comprising a step of using a selection criteria to select a portion of said modules to apply said at least one second genetic operator to, and wherein said at least one second genetic operator includes one or more of crossover and mutation.

24. A method as defined by claim 21 wherein said termination condition is a first termination condition, and wherein the step of applying at least one second genetic operator to said population of variables on a module-specific basis is followed by a step of determining whether a second termination condition has been met, and of repeating said step of applying at least one second genetic operator to said population of variables on a module-specific basis until said second termination condition is met.

25. A method as defined by claim 1 and further including a step of using said optimal design structure matrix clustering to organize a population of variables into modules, and of applying crossover to either all or none of said variables within each of said modules.

26. A method for organizing variables into modules, comprising the steps of:

developing a design structure matrix from the population of variables;

optimizing clustering of the design structure matrix through iterative steps of applying at least one genetic operator to a parent population of design structure matrix clusterings to generate an offspring population of

design structure matrix clusterings until a first termination condition is met and an optimal clustering is defined;

use said optimal clustering to create modules of variables; and,

iteratively apply at least one genetic operator to selected ones of said modules of variables until a second termination condition is met.

27. A computer program product for creating modules of variables, the program product comprising computer executable instructions stored on a computer readable medium that when executed cause a computer to:

organize the variables into a design structure matrix;

create a parent population of design structure matrix clusterings;

apply at least one genetic operator to said parent population of design structure matrix clusterings to produce an offspring population of design structure matrix clusterings;

use a scoring metric to score said offspring population of design structure matrix clusterings;

define an optimal clustering if a termination condition has been satisfied and if no termination condition has been satisfied perform a selection operation to create a new parent population and repeat the steps of applying at least one genetic operator and using a scoring metric until said termination condition has ebbeen achieved;

use said optimal clustering to define modules of the variables; and,

apply at least one genetic operator on a module-specific basis to selected ones of said modules to generate offspring modules.

28. A computer program product as defined by claim 27 wherein the program instructions when executed further cause the computer to:

evaluate said offspring modules of said variables to determine if a second termination condition has been satisfied, and if no second termination

condition has been satisfied to repeat the step of applying at least one genetic operator to selected ones of said modules on a module-specific basis to create new offspring modules until said second termination condition has been met.